

We claim:

5 1. A method for generating an intermediate representation of program code written for running on a programmable machine, said method comprising:

(i) generating a plurality of register objects for  
10 holding variable values to be generated by the program code; and

(ii) generating a plurality of expression objects representing fixed values and/or relationships between  
15 said fixed values and said variable values according to said program code;

wherein at least one variable sized register is represented by plural register objects, one register  
20 object being provided for each possible size of the variably sized register.

2. A method according to claim 1, wherein a write operation to a variably sized register is effected by  
25 writing to the register object corresponding to the appropriate size and maintaining a record of which register objects contain valid data.

3. A method according to claim 2, wherein a read  
30 operation from a variably sized register is effected by determining from said record if there is valid data in more than one corresponding register object which must be

combined to give the same effect as reading from the  
variably sized register, and

(i) if it is determined that no such combination  
5 is required, reading from the appropriate register object;  
and

(ii) if it is determined that such combination is  
required, combining the contents of appropriate register  
10 objects to provide a read value.

4. The method of claim 1, comprising translating the  
program code written for execution by a processor of a  
first type so that the program code may be executed by a  
15 processor of a second type, using the generated  
intermediate representation.

5. The method of claim 4, wherein the translation is  
performed dynamically as the program code is run.  
20

6. The method of claim 1, comprising optimising the  
program code by optimising said generated intermediate  
representation.

25 7. The method of claim 6, wherein the optimising step  
is used to optimise the program code written for execution  
by a processor of a first pipe so that the program code  
may be executed more efficiently by that processor.

30 8. A method of generating an intermediate  
representation of program code expressed in terms of the  
instruction set of a subject processor comprising at least

one variable sized register, the method comprising the computer implemented steps of:

generating a set of associated abstract register  
5 objects representing the variable sized register;

for each write operation of a certain field width to  
the variable sized register, writing to an abstract  
register of the same width;

10

maintaining a record of which abstract register  
objects contain valid data, which record is updated upon  
each write operation; and

15 for each read operation of a given field width,  
determining from said record whether there is valid data  
in more than one of said different sized abstract  
registers of the set which must be combined to give the  
same effect as the same read operation performed upon the  
20 variable size register; and

(a) if it is determined that no combination is so  
required, reading directly from the appropriate register;  
or

25

(b) if it is determined that data from more than  
one register must be so combined, combining the contents  
of those registers.

30 9. The method according to claim 4, wherein the step  
of determining whether or not the contents of more than  
one abstract register must be combined and if so which  
abstract registers must be combined, is determined in

accordance with the following conditions in respect of each set of different sized abstract registers:

(i) if the data required for an access lies wholly  
5 within one valid abstract register, that register only is accessed; and

(ii) if the data required for an access lies within  
more than one valid abstract register, data is combined  
10 from those valid abstract registers to perform the access.

10. A system for generating an intermediate representation of program code written for running on a programmable machine, the system comprising:

15 means for generating a plurality of register objects for holding variable values to be generated by the program code; and

20 means for generating a plurality of expression objects representing fixed values and/or relationships between said fixed values and said variable values according to said program code;

25 wherein at least one variably sized register is represented by plural register objects, one register object being provided for each possible size of the variably sized register.

30 11. A system for generating an intermediate representation of program code expressed in terms of the instruction set of a subject processor comprising of at least one variably sized register, the system comprising:

means for generating a set of associated abstract register objects representing the variably sized register;

5 means for writing, for each write operation of a certain field width to the variable sized register to an abstract register object of the same width;

means for maintaining a record of which abstract  
10 register objects contain valid data, the record being updated upon each write operation; and

means for determining from said record, for each read  
15 operation of a given width, whether there is valid data in more than one of said different sized abstract registers of the set which must be combined to give the same effect as the same read operation performed upon the variable size register, and

20 (a) if it is determined that no combination is so required, reading directly from the appropriate register; or

(b) if it is determined that data from more than  
25 one register must be so combined, combining the contents of those registers.

30